

AMENDMENTS TO THE CLAIMS

The listing of claims will replace all prior versions, and listings, of claims in the application:

Listing of Claims:

1. (Currently Amended) A computer implemented method for asymmetrically multithreading tasks in a computer environment that includes a plurality of dissimilar processors, said method comprising:

retrieving an application;

analyzing a first task type and a second task type included in the application;

in response to analyzing the first task type and the second task type, determining that the application requires both a first scheduler and a second scheduler, wherein the first scheduler is different than the second scheduler;

scheduling, during execution of the application, a plurality of first tasks adapted to be executed by a first type of processor from the plurality of dissimilar processors, the scheduling performed by the first scheduler that maintains a first run queue that includes data corresponding to the first tasks, the plurality of first tasks corresponding to the first task type;

in response to the scheduling of the plurality of first tasks by the first scheduler, inserting the plurality of first tasks in the first run queue;

scheduling, during the execution of the application, a plurality of second tasks adapted to be executed by a second type of processor from the plurality of dissimilar processors, the scheduling performed by the second scheduler that maintains a second run queue that includes data corresponding to the second tasks, wherein the first scheduler and the second scheduler are both utilized by the application during the application execution, and wherein the first task and the second task execute concurrently, the plurality of second tasks corresponding to the second task type;

in response to the scheduling of the plurality of second tasks by the second scheduler, inserting the plurality of second tasks in the second run queue;

informing the second type of processor to load one of the plurality of second tasks in response to the scheduling of the plurality of second tasks; and

wherein the first scheduler and the second scheduler are both located on a first processor that is the first type of processor.

2. (Canceled)
3. (Previously Presented) The method as described in claim 1 wherein the scheduling operation of the plurality of first tasks is asymmetric to the scheduling operation of the plurality of second tasks.
4. (Previously Presented) The method as described in claim 1 further comprising:
wherein a first identifier space corresponds to the first type of processor and
wherein a second identifier space corresponds to the second type of processor;
and
wherein a first task list corresponds to the first type of processor and wherein a second task list corresponds to the second type of processor.
5. (Previously Presented) The method as described in claim 1 wherein the scheduling operation of the plurality second tasks further comprises:
receiving a new task from the plurality of second tasks;
identifying new task attributes corresponding to the new task;
comparing the new task attributes with one or more scheduled task attributes, the scheduled task attributes corresponding to one or more scheduled tasks that are included in the second run queue; and
performing the scheduling of the new task based upon the comparing.

6. (Original) The method as described in claim 5 wherein at least one of the new task attributes are selected from the group consisting of a policy and a priority.
7. (Canceled)
8. (Original) The method as described in claim 1 wherein the computer environment includes a plurality of second type of processors, and wherein the second scheduler maintains a plurality of second run queues, each of the plurality of second run queues corresponding to each of the plurality of second type of processors.
9. (Original) The method as described in claim 8 wherein the scheduling operation of the plurality of second tasks further comprises:
receiving a new task from the plurality of second tasks
analyzing a plurality of workloads that correspond to the plurality of second type of processors;
identifying an available second type of processor from the plurality of second type of processors in response to the analyzing; and
including the new task in the second run queue from the plurality of second run queues that corresponds to the available second type of processor.
10. (Previously Presented) The method as described in claim 1 wherein the first type of processor is a processing unit that operates in a trusted mode environment and wherein the second type of processor is a synergistic processing unit that operates in a non-trusted mode environment.
11. (Canceled)
12. (Canceled)

13. (Canceled)

14. (Canceled)

15. (Canceled)

16. (Canceled)

17. (Canceled)

18. (Canceled)

19. (Canceled)

20. (Canceled)